



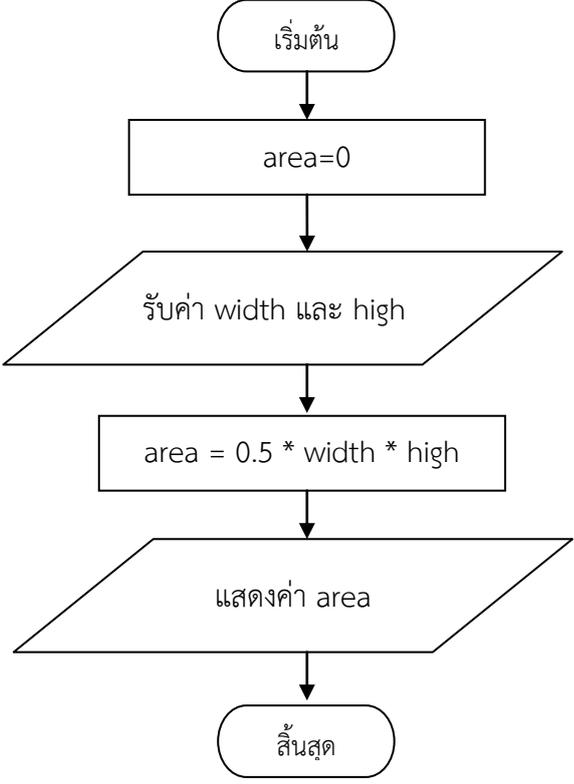
## 3.2 การวิเคราะห์โจทย์ปัญหา

การวิเคราะห์โจทย์ปัญหา เป็นกระบวนการแก้ปัญหที่สามารถเข้าใจได้ มีลำดับหรือวิธีการในการแก้ไขปัญหาใดปัญหาหนึ่งอย่างเป็นขั้นเป็นตอนและชัดเจน เมื่อนำเข้าอะไร แล้วจะต้องได้ผลลัพธ์อะไร ในกระบวนการวิเคราะห์โจทย์ปัญหานี้จะรวมไปถึงการออกแบบการทำงานของโปรแกรมและหน้าจอภาพที่ใช้สำหรับติดต่อกับผู้ใช้งานด้วย ซึ่งจะมีขั้นตอนในการวิเคราะห์ 7 ขั้นตอน ดังนี้

1. การวิเคราะห์ผลลัพธ์ เป็นการวิเคราะห์ว่าปัญหาต้องการผลลัพธ์เป็นอะไร เช่น เป็นตัวเลข ตัวอักษร หรือข้อความ
2. การวิเคราะห์ข้อมูลนำเข้า เป็นการวิเคราะห์ถึงข้อมูลและแหล่งที่มาของข้อมูลที่จะนำเข้ามาประมวลผล
3. การวิเคราะห์การประมวลผล เป็นการวิเคราะห์ถึงวิธีการประมวลผลข้อมูล ที่จะทำให้ได้ผลลัพธ์ที่ต้องการ เช่น การคำนวณ การเปรียบเทียบเพื่อเลือกทิศทาง
4. การวิเคราะห์ตัวแปร เป็นเปลี่ยนข้อมูลนำเข้าและผลลัพธ์ให้เป็นตัวแปรที่แทนด้วยอักษรภาษาอังกฤษให้พร้อมสำหรับนำไปเขียนโปรแกรม
5. การวิเคราะห์ขั้นตอนในการทำงาน เป็นการเขียนบรรยายขั้นตอนการทำงานที่ละขั้นตอนตั้งแต่เริ่มต้นจนถึงสิ้นสุดโปรแกรม
6. การวิเคราะห์แผนผังลำดับงาน เป็นการใช้สัญลักษณ์วาดแผนผังแทนขั้นตอนการทำงานที่เขียนบรรยายในขั้นตอนที่ 5
7. การวิเคราะห์การออกหน้าจอ เป็นการออกแบบหน้าจอโปรแกรมเป็นการวางแผนการวางองค์ประกอบของคอนโทรลต่างๆ บนหน้าจอเพื่อประหยัดเวลาในการเขียนโปรแกรม

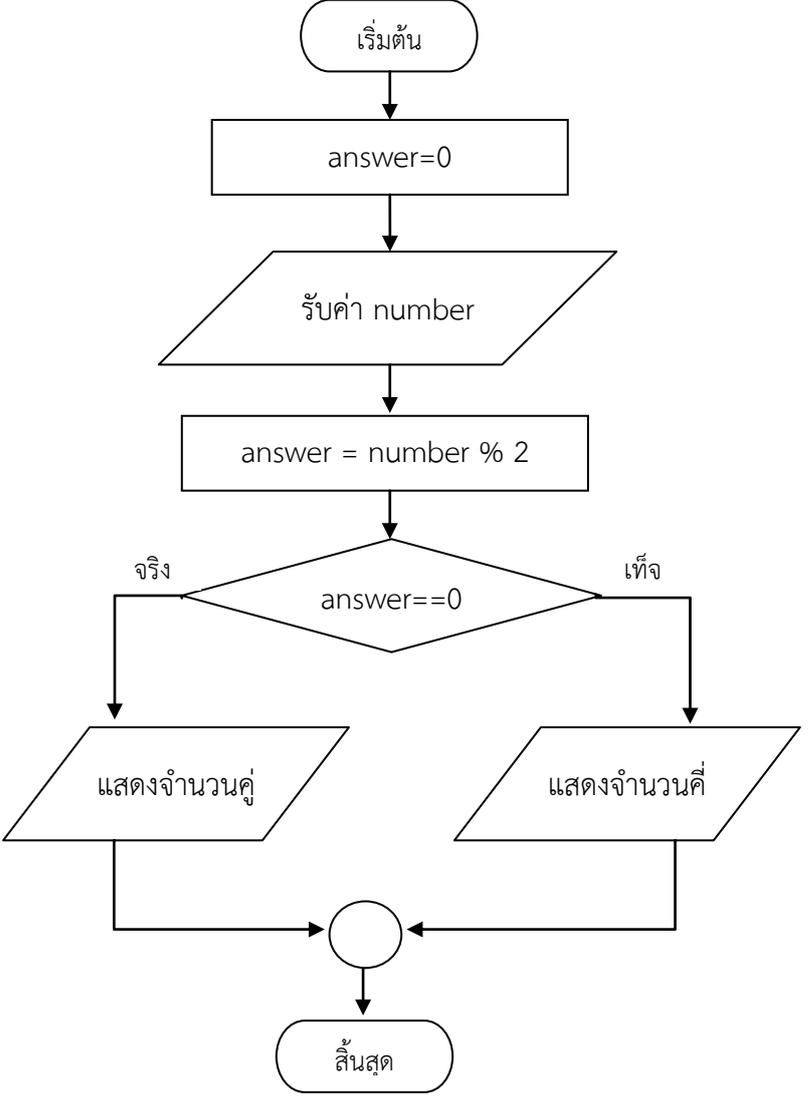
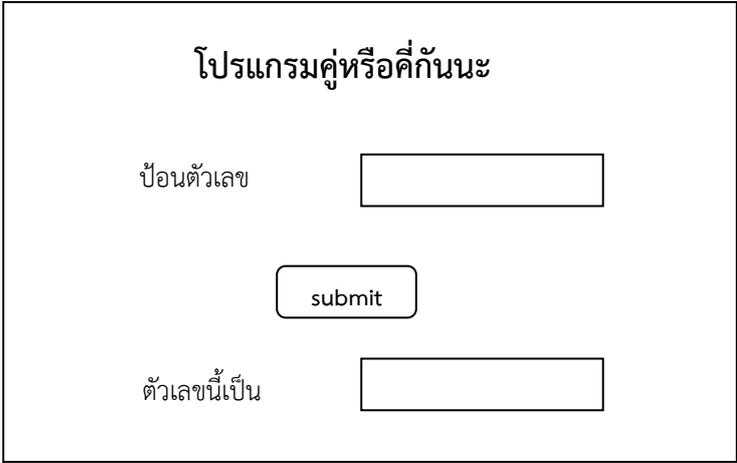
**ตัวอย่างที่ 1** : จงวิเคราะห์โจทย์ปัญหาการคำนวณหาพื้นที่สามเหลี่ยม โดยรับข้อมูลความกว้างของฐาน และความสูงของสามเหลี่ยมจากผู้ใช้

ขั้นตอนการวิเคราะห์โจทย์ปัญหา	ผลการวิเคราะห์
1. ผลลัพธ์	พื้นที่สามเหลี่ยม
2. ข้อมูลนำเข้า	1. ความกว้างของฐาน 2. ความสูงของสามเหลี่ยม
3. การประมวลผล	พื้นที่สามเหลี่ยม = $\frac{1}{2} * \text{ความกว้างของฐาน} * \text{ความสูงของสามเหลี่ยม}$
4. ตัวแปร	1. พื้นที่สามเหลี่ยม แทนด้วย area 2. ความกว้างของฐาน แทนด้วย width 3. ความสูงของสามเหลี่ยม แทนด้วย high
5. ขั้นตอนการทำงาน	5.1 เริ่มต้น 5.2 กำหนดให้ พื้นที่สามเหลี่ยม (area) มีค่าเริ่มต้นเท่ากับ 0 5.3 รับค่า ความกว้างของฐาน (width) และ ความสูงของสามเหลี่ยม (high) 5.4 ประมวลผลเพื่อหาผลลัพธ์ ดังนี้ พื้นที่สามเหลี่ยม = $\frac{1}{2} * \text{ความกว้างของฐาน} * \text{ความสูงของสามเหลี่ยม}$ หรือ area = $0.5 * \text{width} * \text{high}$ 5.5 แสดงพื้นที่สามเหลี่ยม(area) 5.6 สิ้นสุดการทำงาน

ขั้นตอนการวิเคราะห์โจทย์ปัญหา	ผลการวิเคราะห์
6. แผนผังลำดับงาน	 <pre> graph TD     Start([เริ่มต้น]) --&gt; Init[area=0]     Init --&gt; Input[/รับค่า width และ high/]     Input --&gt; Calc[area = 0.5 * width * high]     Calc --&gt; Output[/แสดงค่า area/]     Output --&gt; End([สิ้นสุด]) </pre>
7. ออกแบบหน้าจอ	<div style="border: 1px solid black; padding: 10px; text-align: center;"> <p><b>โปรแกรมพื้นที่สามเหลี่ยม</b></p> <p>ป้อนความกว้างของฐาน <input style="width: 100px;" type="text"/></p> <p>ป้อนความสูง <input style="width: 100px;" type="text"/></p> <p><input type="button" value="submit"/></p> <p>พื้นที่สามเหลี่ยมเท่ากับ <input style="width: 100px;" type="text"/></p> </div>

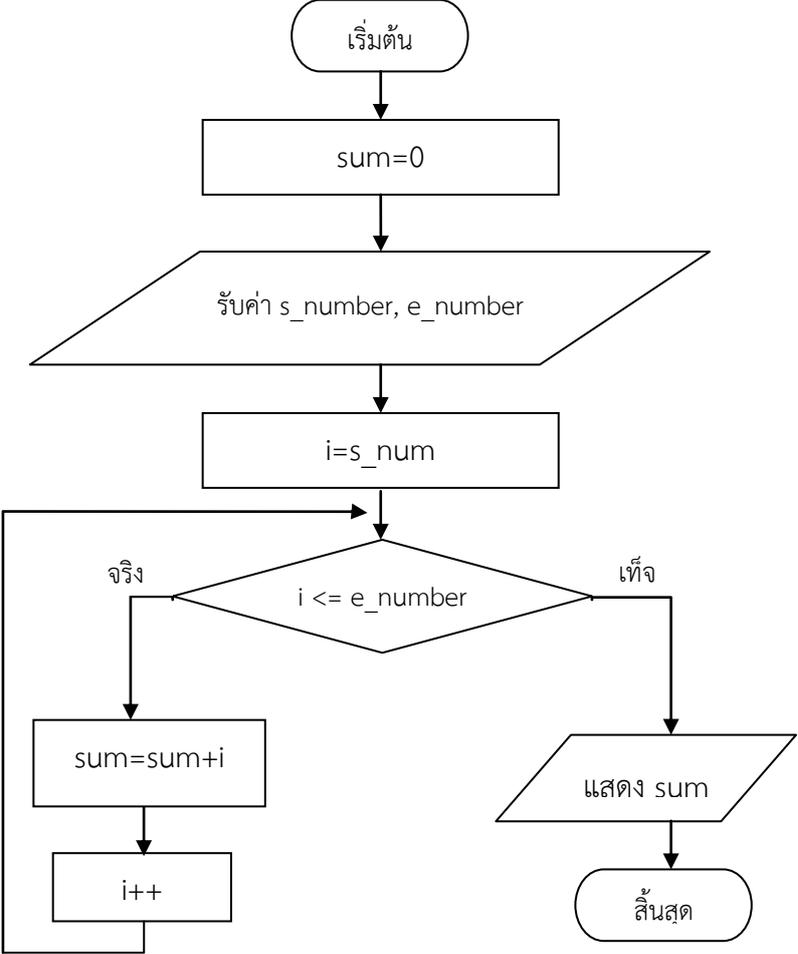
**ตัวอย่างที่ 2** : จงวิเคราะห์โจทย์ปัญหาการคำนวณหาจำนวนคู่หรือจำนวนคี่จากการป้อนตัวเลข 1 จำนวน

ขั้นตอนการวิเคราะห์โจทย์ปัญหา	ผลการวิเคราะห์
1. ผลลัพธ์	จำนวนคู่หรือจำนวนคี่
2. ข้อมูลนำเข้า	ตัวเลข 1 จำนวน
3. การประมวลผล	เศษที่ได้ = ตัวเลข 1 จำนวน / 2
4. ตัวแปร	1. เศษที่ได้ แทนด้วย answer 2. ตัวเลข 1 จำนวน แทนด้วย number
5. ขั้นตอนการทำงาน	5.1 เริ่มต้น 5.2 กำหนดให้ เศษที่ได้ (answer) มีค่าเริ่มต้นเท่ากับ 0 5.3 รับค่า ตัวเลข 1 จำนวน (number) 5.4 ประมวลผลเพื่อหาผลลัพธ์ ดังนี้ เศษที่ได้ = ตัวเลข 1 จำนวน%2 หรือ answer = number / 2 5.5 นำเศษที่ได้มาเปรียบเทียบ ดังนี้ 1. ถ้าเศษที่ได้(answer)เท่ากับ 0 : ให้แสดงข้อความว่า เป็นจำนวนคู่ 2. ถ้าเศษที่ได้(answer)เท่ากับ 1 : ให้แสดงข้อความว่า เป็นจำนวนคี่ 5.6 สิ้นสุดการทำงาน

ขั้นตอนการวิเคราะห์โจทย์ปัญหา	ผลการวิเคราะห์
6. แผนผังลำดับงาน	 <pre> graph TD     Start([เริ่มต้น]) --&gt; Init[answer=0]     Init --&gt; Input[/รับค่า number/]     Input --&gt; Calc[answer = number % 2]     Calc --&gt; Decision{answer==0}     Decision -- จริง --&gt; OutputEven[/แสดงจำนวนคู่/]     Decision -- เท็จ --&gt; OutputOdd[/แสดงจำนวนคี่/]     OutputEven --&gt; Merge(( ))     OutputOdd --&gt; Merge     Merge --&gt; End([สิ้นสุด]) </pre>
7. ออกแบบหน้าจอ	

**ตัวอย่างที่ 3** : จงวิเคราะห์โจทย์ปัญหาการบวกเลขโดยผู้ใช้งานต้องระบุตัวเลขเริ่มต้น และตัวเลขสิ้นสุด ซึ่งมีค่าอยู่ระหว่าง 1-100

ขั้นตอนการวิเคราะห์โจทย์ปัญหา	ผลการวิเคราะห์
1. ผลลัพธ์	ผลบวก
2. ข้อมูลนำเข้า	1. ตัวเลขเริ่มต้น 2. ตัวเลขสิ้นสุด
3. การประมวลผล	ผลบวก = ตัวเลขเริ่มต้น + (ตัวเลขเริ่มต้น+1) + (ตัวเลขเริ่มต้น+2) +.....+ ตัวเลขสิ้นสุด
4. ตัวแปร	1. ผลบวก            แทนด้วย sum 2. ตัวเลขเริ่มต้น    แทนด้วย s_num 3. ตัวเลขสิ้นสุด      แทนด้วย e_num 4. รอบในการบวก    แทนด้วย i
5. ขั้นตอนการทำงาน	5.1 เริ่มต้น 5.2 กำหนดให้ ผลบวก (sum) มีค่าเริ่มต้นเท่ากับ 0 5.3 รับค่า ตัวเลขเริ่มต้น (s_num) และ ตัวเลขสิ้นสุด (e_num) 5.4 กำหนดให้การทำงาน - เริ่มต้นรอบการบวกที่ ตัวเลขเริ่มต้น (i=s_num) - ตรวจสอบว่ายังไม่เกินตัวเลขสิ้นสุด - ตัวเลขเริ่มต้นบวกไปที่ละ 1 จนกว่าจะถึงตัวเลขสิ้นสุด 5.5 แสดงผลบวก (sum) 5.6 สิ้นสุดการทำงาน

ขั้นตอนการวิเคราะห์โจทย์ปัญหา	ผลการวิเคราะห์
6. แผนผังลำดับงาน	 <pre> graph TD     Start([เริ่มต้น]) --&gt; Sum0[sum=0]     Sum0 --&gt; Input[/รับค่า s_number, e_number/]     Input --&gt; IAssign[i=s_num]     IAssign --&gt; Loop{i &lt;= e_number}     Loop -- จริง --&gt; SumAdd[sum=sum+i]     SumAdd --&gt; IInc[i++]     IInc --&gt; Loop     Loop -- เท็จ --&gt; Display[/แสดง sum/]     Display --&gt; End([สิ้นสุด]) </pre>
7. ออกแบบหน้าจอ	<div style="border: 1px solid black; padding: 10px; text-align: center;"> <p><b>โปรแกรมบวกเลขแบบมีเงื่อนไข</b></p> <p>ป้อนตัวเลขเริ่มต้น <input style="width: 100px;" type="text"/></p> <p>ป้อนตัวเลขสิ้นสุด <input style="width: 100px;" type="text"/></p> <p><input type="button" value="submit"/></p> <p>ผลบวกเท่ากับ <input style="width: 100px;" type="text"/></p> </div>